Badge

Badge A smart notification badge component for displaying counts, status indicators, and priority levels. Features intelligent number formatting (9+, 99+, 999+), multiple size variants including dots, semantic color states, and comprehensive accessibility support for enhanced user communication. How to use ■ import { AavaBadgesComponent } from "@aava/play-core" ; Basic Usage ■ Simple badge implementations with counts and basic styling. Sizes ■ Four size variants to fit different interface requirements and visual hierarchy. Available Sizes ■ xs (Extra Small) - Minimal size for very dense interfaces and subtle indicators sm (Small) - Compact for dense interfaces and subtle indicators md (Medium) - Standard size for most use cases (default) lg (Large) - Prominent for important notifications and better accessibility Variants ■ Three variant types to serve different use cases. Available Variants ■ Default Icon Dots Accessibility ■ Built-in accessibility features ensuring inclusive user experience. Accessibility Features ■ Keyboard Navigation : Tab navigation and keyboard activation Screen Reader Support : Semantic HTML and ARIA attributes Focus Indicators : Clear visual focus states for navigation Color Independence : Information conveyed beyond color alone High Contrast : Enhanced visibility in high contrast modes Descriptive Content : Meaningful count and status information API Reference ■ Inputs ■ Property Type Default Description state 'high-priority' | 'medium-priority' | 'low-priority' | 'neutral' | 'information' | 'online' | 'offline' 'neutral' Semantic state determining badge color size 'lg' | 'md' | 'sm' | 'xs' 'md' Size variant of the badge variant 'default' | 'dots' 'default' Visual variant of the badge count number undefined Number to display (smart formatting applied) iconName string undefined Name of icon to display (from ava-icon) iconColor string 'white' Custom color for the icon iconSize number undefined Size of the icon in pixels customStyles Record<string, string> Properties ■ Property Type Description displayCount string Formatted count string (includes 9+, 99+, etc.) badgeClasses string Computed CSS classes for badge styling hasContent boolean Whether badge has count or icon content isDots boolean Whether badge is dots variant isSingleDigit boolean Whether count is a single digit (0-9) Methods ■ Method Parameters Return Type Description onKeyPress() - void Handle keyboard interaction (Enter/Space) CSS Custom Properties ■ Property Description --badge-font Font for badge text --badge-weight Font weight for badge text --badge-line-height Line height for badge text --badge-border-radius Border radius for badge --badge-padding Padding inside badge --badge-display CSS display property for badge --badge-align-items Vertical alignment of badge content --badge-justify-content Horizontal alignment of badge content --badge-gap Gap between badge elements --badge-default-background Background color for neutral state --badge-default-text Text color for neutral state --badge-default-border Border for neutral state --badge-primary-background Background color for primary state --badge-primary-text Text color for primary state --badge-primary-border Border for primary state --badge-secondary-background Background color for secondary state --badge-secondary-text Text color for secondary state --badge-secondary-border Border for secondary state --badge-success-background Background color for success state --badge-success-text Text color for success state --badge-success-border Border for success state --badge-warning-background Background color for warning state --badge-warning-text Text color for warning state --badge-warning-border Border for warning state --badge-error-background Background color for error state --badge-error-text Text color for error state --badge-error-border Border for error state --badge-info-background Background color for information state --badge-info-text Text color for information state --badge-info-border Border for information state --badge-online-background Background color for online state --badge-offline-background Background color for offline state --badge-size-xs-min-width Minimum width for extra small badge --badge-size-xs-height Height for extra small badge --badge-size-sm-font Font for small badge --badge-size-sm-padding Padding for small badge --badge-size-sm-min-width Minimum width for small badge --badge-size-sm-height Height for small badge --badge-size-md-font Font for medium badge --badge-size-md-padding Padding for medium badge --badge-size-md-min-width Minimum width for medium badge --badge-size-md-height Height for medium badge --badge-size-lg-font Font for large badge --badge-size-lg-padding Padding for large badge --badge-size-lg-min-width Minimum width for large badge --badge-size-lg-height Height for large badge --badge-solid-background Background color for solid variant --badge-solid-text Text color for solid variant --badge-solid-border Border for solid variant --badge-outline-background Background color for outline variant --badge-outline-text Text color for outline variant --badge-outline-border Border for outline variant --badge-ghost-background Background color for ghost variant --badge-ghost-text Text color for ghost variant --badge-ghost-border Border for ghost variant --badge-dot-size Size of dot indicator --badge-dot-border-radius Border radius for dot (circle) --badge-dot-margin-right Margin right for dot --badge-icon-size Default size for icons --badge-icon-color Default icon color --badge-icon-margin-right Margin right for icons --badge-counter-background Background color for counter badges --badge-counter-text Text color for counter badges --badge-counter-font Font for counter badges --badge-counter-min-width Minimum width for counter badges --badge-counter-height Height for counter badges --badge-counter-border-radius Border radius for counter badges --badge-counter-padding Padding for counter badges --badge-font-family Font family for badge text Best Practices ■ Design Guidelines ■ Use high-priority for urgent notifications requiring immediate attention Choose appropriate sizes based on interface density and importance Prefer neutral state for simple count displays Use dots variant for minimal status indicators without text Use icons for status indicators rather than counts Consider badge placement to avoid blocking important content Accessibility ■ Ensure badges convey information beyond color alone Provide meaningful text content for screen readers Use appropriate state colors that meet contrast requirements Test keyboard navigation thoroughly Consider announcing dynamic count changes to screen readers Performance ■ Avoid frequent count updates that trigger excessive re-renders Use OnPush change detection strategy for optimal performance Consider debouncing rapid count changes Cache computed properties for better performance Technical Notes ■ Automatic Expansion ■ The badge automatically expands its width when displaying multi-character content (like "99+" or "999+") while maintaining circular dimensions for single characters. Icon vs Count Priority ■ When both iconName and count are provided, the count takes priority and the icon is not displayed. Use separate badges for icon + count combinations. Dots Variant ■ The dots variant creates simple circular indicators without text content. It's ideal for status indicators, online/offline states, or any minimal visual cue that doesn't require text. Keyboard Interaction ■ Badges with tabindex="0" can receive keyboard focus and respond to Enter and Space key presses for custom interactions. The component uses the selector ava-badges (plural) for consistency with the component library naming convention.

```
<aava-badges
  [count]="5"
  [state]="'neutral'"
  [size]="'md'"
  (click)="onBadgeClick($event)"
  (keydown.enter)="onBadgeKeyPress($event)"
  (keydown.space)="onBadgeKeyPress($event)"
  tabindex="0"
  role="button"
  aria-label="Badge with count 5"
>
</aava-badges>

<aava-badges
  [count]="3"
  [state]="'high-priority'"
  [size]="'md'"
  (click)="onBadgeClick($event)"
  (keydown.enter)="onBadgeKeyPress($event)"
  (keydown.space)="onBadgeKeyPress($event)"
  tabindex="0"
  role="button"
  aria-label="Badge with count 3"
>
</aava-badges>

<aava-badges
  [count]="12"
  [state]="'low-priority'"
  [size]="'md'"
  (click)="onBadgeClick($event)"
  (keydown.enter)="onBadgeKeyPress($event)"
  (keydown.space)="onBadgeKeyPress($event)"
  tabindex="0"
  role="button"
  aria-label="Badge with count 12"
>
</aava-badges>

<aava-badges
  [count]="8"
  [state]="'information'"
  [size]="'md'"
  (click)="onBadgeClick($event)"
  (keydown.enter)="onBadgeKeyPress($event)"
  (keydown.space)="onBadgeKeyPress($event)"
  tabindex="0"
  role="button"
  aria-label="Badge with count 8"
>
</aava-badges>


---

  onBadgeClick(event: Event): void {
```

```
  console.log('Badge clicked:', event);
}

onBadgeKeyPress(event: Event): void {
  console.log('Badge key pressed:', event);
}
```

```
<aava-badges
  [count]="1"
  [state]="'high-priority'"
  [size]="'xs'"
  (click)="onBadgeClick($event)"
  tabindex="0"
  role="button"
  aria-label="Badge with count 1 and xs size"
>
</aava-badges>

<aava-badges
  [count]="5"
  [state]="'medium-priority'"
  [size]="'sm'"
  (click)="onBadgeClick($event)"
  tabindex="0"
  role="button"
  aria-label="Badge with count 5 and sm size"
>
</aava-badges>

<aava-badges
  [count]="15"
  [state]="'low-priority'"
  [size]="'md'"
  (click)="onBadgeClick($event)"
  tabindex="0"
  role="button"
  aria-label="Badge with count 15 and md size"
>
</aava-badges>

<aava-badges
  [count]="99"
  [state]="'information'"
  [size]="'lg'"
  (click)="onBadgeClick($event)"
  tabindex="0"
  role="button"
  aria-label="Badge with count 99 and lg size"
>
</aava-badges>


---

  onBadgeClick(event: Event): void {
    console.log('Badge clicked:', event);
  }
```

■ No code found